Jerzy Mycka
University of Maria Curie-Sklodowska, Lublin

EMPIRICAL ASPECTS OF COMPUTABILITY THEORY
that is: does the Univers compute better than (thoughtless) man?

Abstract. The paper deals with the question how the shape of physical reality affects the classification of problems into computable and noncomputable. First A. Turing's idea of computability is recalled. On the basis of this theory a possibility of revising the notion of computability will be presented. The starting point will be a conception of a physical system processing information understood as parameters of the system elements. Appropriate for this approach models of computability which are not brought to Turing machines will be pointed out. As the major objection to the models of the above-mentioned type the Gandy thesis will be considered. The paper justifies methodologically as plausible the search of the problems exceeding, in the sense of computability, the Turing model.

Computability

The idea of mechanical solving of problems goes back to the distant past in science. We will not, however, deal with the historical outline of this issue, but immediately turn to the notions accompanying the examinations of computability which were coined by Alan Turing [18].

The model of computability called the Turing machine, described by means of an informal language, can be defined as follows: the machine comprises an infinite tape divided into identical cells, which is to store input data, output data and working information. All the elements on the tape are strings (sequences of symbols); at the same time there is a rule of placing one symbol into one cell. Without a loss of generality a particular alphabet is usually chosen as a range of symbols permissible to build strings. Practically the binary (zero, one) alphabet is often chosen, it allows a convenient and consistent representation of data. Moreover, the construction of the machine requires a description of a finite set of states from which an element, indicating a current situation (state) of the machine, originates.

ISBN 83-7431-015-4 ISSN 0860-150X

The Turing machine works in steps which are identical as to the time of lasting. At every step the machine reads the content of a current cell (pointed out by the head), then it changes the content according to a read symbol and a present state. The same information serves successively to change the state and to move the head to the next left or right square. Particular states are distinguished as final; if the machine is in one of them, it finishes its work. The classical model of the Turing machine requires the number of symbols on the tape (different from the empty symbol) to be finite at every moment of the machine work.

A problem is computable, according to A. Turing, if its solution can be found due to the properly constructed Turing machine. Let us notice that the whole process of computing described above can be easily imagined as the work of a man who, by means of a sheet of paper and a pencil, realizes (thoughtlessly) consecutive changes of symbols according to strict rules. Robert Soare [17] stressed the character of Turing's computability, naming the computing subject 'computor', to emphasize the idealization of human activity which was used here. An analysis of the machine potentials (physical systems) does not seem to be the Turing's aim. In agreement with the previous remarks the construction of Turing machines encloses rather the possibilities of 'an ideal mathematician' activity.

Turing machines defined in the following way allow a comprehensive analysis of computability problems. The most important are: distinguishing the issues that cannot be explained by means of the Turing machine (unsolvable problems) and pointing out time and spatial restrictions originating from the nature of a solved issue (theory of complexity).

Of course, the Turing machine is not the only one model of computable processes. Just to illustrate, there are the Markov algorithms [6], the Church λ-calculus [1] or partially recursive functions all of which were proposed independently. In the course of research for all the models the thesis has been verified that they are identical as far as the scope of computability is concerned (compare [10]). Using an informal language: every issue explained in one of these models can be solved in any other model as well.

Let us digress from the subject here. The models are equivalent as to the power but different as regards a means of expression. The Markov algorithms and the Church calculus are the models based on the description of particular operations on strings. Nevertheless, it is an approach which is very close to the one usually accepted by adherents of formalism, regarding the foundations of mathematics. Then the description of calculations reflecting human activities – the Turing machine – leads us into the areas that are close to mathematical intuitionism. The world of mathematical functions in the natural domain, yet, seems to be the domain of platonism. As it can be suggested by this short enumeration, a preference for using one computability model may be connected with philosophical orientation regarding the foundations of mathematics.

The results of researches concerning different computability models led to formulate the Church-Turing thesis [2]. It will be recalled at this stage in order to avoid misunderstandings. The above thesis claims that our informal notion of an effective computing is parallel to the precise Turing machine mathematical definition. The conditions usually associated with efficiency are connected with a finite number of activity instructions (rules) which are represented by finite strings accompanied by action (in a proper case) in a finite number of steps with a possibility (in general) of realizing the algorithm by a man without using intuition, creativity, or direct insight into the essence. Let us pay attention to what the Church-Turing thesis does not say: it does not claim that other types of computing are not possible. It just points out that the calculations that are effective are related to the above mentioned models.

Hypercomputabilty

In the light of the above mentioned facts the question arises, essential both theoretically and practically – is there a form of an ineffective calculation, yet possible to be realized? Ordinary procedures and models of this property are described by the notion of hypercomputability. Since an effectiveness of calculations seems to represent a human activity description by means of the theory of computability language, thus ineffective solutions are sought in the world of physics.

In agreement with these remarks we will present the possibilities of revising the notion of computability. The starting point will be a concept of physical (artificial or natural) system transferring the information perceived as parameters of elements of the system. A mechanical character of such an approach agrees with a traditional name ‘computer’.

Let us begin from introducing examples of ineffective computability models. A straightforward modification of the Turing machine, called the accelerating Turing machine, will be presented as the first model. The description of its structure is identical to the basic Turing machine. The change concerns the timing pattern of realized steps. Every consecutive step is realized in the time equal to the half time of the preceding step. The machine working in the following way is able to produce an infinite number of steps during the first two initial units of time. This feature allows solving the problem of halting some Turing machine by the following widening of the set of its instructions: firstly, mark with 0 symbol the chosen cell of the machine; in case of introducing the final instruction – change the input of this cell into 1. After two units of time an examination of this cell allows us to state whether the Turing machine stopped after finite or infinite number of steps. Because the halting problem is not effectively computable, a new model spreads significantly the limits of the notion of computability.

Let us turn into the other field of nonefective computation: namely analog computation. The basic model in this field is Shannon's General Purpose Analog Computer [16].

The General Purpose Analog Computer (GPAC) is a computer whose computation evolves in continuous time. The outputs are generated from the inputs by means of a dependence defined by a finite directed graph (not necessarily acyclic) where each node is one of the following boxes.

- *Integrator*: a two-input, one-output unit with a setting for initial condition. If the inputs are unary functions u, v, then the output is the Riemann-Stieljes integral $\lambda t.\int_{t_0}^{t} u(x)dv(x) + a$, where a and t_0 are real constants defined by the initial settings of the integrator.
- *Constant multiplier*: a one-input, one-output unit associated to a real number. If u is the input of a constant multiplier associated to the real number k, then the output is ku.
- *Adder*: a two-input, one-output unit. If u and v are the inputs, then the output is $u + v$.
- *Multiplier*: a two-input, one-output unit. If u and v are the inputs, then the output is uv.
- *Constant function*: a zero-input, one-output unit. The value of the output is always 1.

The next important in this context model of analog computation is Rubel's Extended Analog Computer (EAC) [13, 14]. This model is similar to the GPAC, but it allows, in addition, other types of units, e.g. units that solve boundary value problems (here we allow several independent variables because Rubel is not seeking any equivalence with existing models). The EAC permits all the operations of ordinary analysis, except the unrestricted taking of limits. The new units add an extended computational power relatively to the GPAC. For example, the EAC can solve the Dirichlet problem for Laplace's equation in the disk and can generate the Γ function (it is known that the GPAC cannot solve these problems [13]).

The model which is similar in the analog realm to classical natural recursive functions is the system of real recursive functions. Here we present a version given in [9] and based on the work of C. Moore'a [7]. The below definition is based on vector operations.

The set of real recursive vectors is generated from the real recursive scalars $0, 1, -1$ and the real recursive projections $I_n^i(x_1, \ldots, x_n) = x_i, 1 \leq i \leq n, n > 0$, by the operators:

1. composition: if f is a real recursive vector with n k-ary components and g is a real recursive vector with k m-ary components, then the vector with n m-ary components $(1 \leq i \leq n)$

$$\lambda x_1 \ldots x_m . f_i(g_1(x_1, \ldots, x_m), \ldots, g_k(x_1, \ldots, x_m))$$

   is real recursive.
2. differential recursion: if f is a real recursive vector with n k-ary components and g is a real recursive vector with n $k+n+1$-ary components, then the vector h of n $k+1$-ary components which is the solution of the Cauchy problem for $1 \leq i \leq n$

$$h_i(x_1, \ldots, x_k, 0) = f_i(x_1, \ldots, x_k),$$

$$\partial_y h_i(x_1, \ldots, x_k, y) = g_i(x_1, \ldots, x_k, y, h_1(x_1, \ldots, x_k, y), \ldots, h_n(x_1, \ldots, x_k, y))$$

   is real recursive whenever h and its derivative are continuous in y on the largest interval containing 0 in which a unique solution exists except for a countable set of isolated points of discontinuity (of its derivative) where only one analytical continuation exists.
3. infinite limits: if f is a real recursive vector with n $k+1$-ary components, then the vectors h, h', h'' with n k-ary components $(1 \leq i \leq n)$

$$h_i\,(x_1, \ldots, x_k) = \lim_{y \to \infty} f_i(x_1, \ldots, x_k, y),$$

$$h'_i\,(x_1, \ldots, x_k) = \liminf_{y \to \infty} f_i(x_1, \ldots, x_k, y),$$

$$h''_i(x_1, \ldots, x_k) = \limsup_{y \to \infty} f_i(x_1, \ldots, x_k, y),$$

   are real recursive, whenever these limits are defined for all $1 \leq i \leq n$.[1]
4. Arbitrary real recursive vectors can be defined by assembling scalar real recursive components.

[1] These concepts are defined in the completion of the real numbers $R \cup \{-\infty, +\infty\}$.

5. If f is a real recursive vector, than each of its components is a real recursive scalar.

Let us discuss the definition carefully. For differential recursion we restrict a domain to an interval of continuity. This will preserve the analiticity of functions in the process of defining. Moreover, this operator gives the same class C^k for a defined function as the given functions come from. This eliminates a possibility of defining such functions as $\lambda x.|x|$.

Let us point out the fact that this definition has as its feature the property of a real recursive computable equation relation. It is not a general case for an analog computation.

From the physical point of view with such definition we are ready to use only a finite amount of energy. We excluded here the possibility of operations on undefined functions: our functions are strict in the meaning that for undefined arguments they are also undefined. But to obtain some interesting functions we should improve the power of our system by an addition of the operators of infinite limits. Let us point out that introducing infinite limits gets discontinuous functions.

Infinity versus computability

If the Turing machine seems to be practically realized in the physical world, the above-mentioned models raise considerable doubts in this respect. They can be described precisely by means of the Gandy thesis. It is usually formulated in the following way: everything that can be computed by discrete deterministic mechanical device, can be calculated by the Turing machine as well. Let us notice that, using the rule of contraposition in this statement, we obtain an equivalent formulation: a problem impossible to be solved by the Turing machine will not be computable by means of any discrete and deterministic device. As the models mentioned above exceed the limits of classical computability, we would have to, on the strength of the above thesis, admit their practical 'non-realizability' in deterministic and discrete world.

Before an attempt of estimating the Gandy thesis let us consider a phenomenon of solving by these models the issues noncomputable in Turing's sense. An accelerating Turing machine is the simplest case for an analysis. Obtaining the message about a finite or infinite number of operations conducted by the machine is not connected with any refined procedure of examining this property. The machine just realizes its functions in a limited or unlimited number of steps. Appearing of results is a consequence of its ability to realize an infinite number of steps in finite time. The power of this model lies in achieving a result due to find a limit in restricted time. The same possibilities appear for real recursive functions and EAC as a result of inscribing them into a construction of infinite limits operators. It is worth noticing that the same reason causes the mentioned models to be ineffective, namely they do not fulfill the requirement of a finite number of computing steps.

However, recognizing principles of the models constructions and the Gandy thesis to be mutually excluding, seems to be premature. Although discretion and determinism of computing are guaranteed in the thesis, an idea of an infinite number steps device is not rejected explicitly. Assuming implicitly such a limitation is connected with a conviction that an infinite number of steps requires infinite time. However, this statement is not at all obvious and is more connected with qualities of the physical world that surrounds us than with an inner structure of the proposed models.

Infinity versus physics

In the light of previous considerations, for establishing the boundaries of practically realized models of computability, the nature of the material world becomes essential. It is important, however, to become aware of an obvious fact that we do not possess a direct knowledge of this quality of the Universe. That is why an analysis of its features and limits always takes place by means of physical theories. These theories become the only way to perceive quantitative relations that occur in the physical world.

Therefore, we face the next boundary of our analyses. We cannot discuss ultimate boundaries of computability, but the limits of computability possibilities that result from a physical theory which is regarded as given. However, it may appear that such a far-reaching claim, namely the postulate of realizing infinity (energy, time) in finite sector of physical reality is not acceptable to every physical theory. To weaken slightly the last sentence, it is possible to restrict the considerations at least to commonly approved (not particularly exotic) physical theories. It occurs, though, that the above assumptions are not true. Two examples of physical theories allowing hypercomputability will be presented at this stage.

The first is the Newton mechanics. In 19th century P. Painlevé together with H. Poincaré proposed a particular analysis of an issue connected with the mechanics of heavenly bodies, that is the question of n-bodies. In the very issue of n-bodies a solution of an equation system of movements for n gravitational interacting bodies is sought. P. Painlevé and H. Poincaré opened a discussion not about the way to discover a particular solution, but about an analysis of qualities of these solutions. There is a crucial question whether there may exist such problem solutions that contain a singularity. The singularity as a solution has the quality such that its equation adopts infinite (not specified) values. It is obvious that a situation of this kind happens when two, from all the described by the problem, bodies collide. Yet the question arises whether the singularity may appear without any collision. The answer to this question was given by Z. Xia [19] in 1992. He claimed that for a problem of five bodies in the three-dimensional space there exist non-collision solutions. The Xia answer causes throwing one of the bodies to infinity in finite time. As it can be seen, the Newton mechanics allows finite realizations of infinity and potentially supports a possibility of calculations exceeding the limits of the Turing machine.

An obvious aim that appears at this moment is relating similar considerations to the physical theories that are regarded as currently valid. For this purpose we will use the theory of general relativity. There are such solutions of Einstein's equations in which there exists a time-like half-curve γ as well as a point p in spacetime such that the entire stretch γ is contained in the chronological past of p. Such spacetime structures (i.e. anti-de Sitter spacetimes) have been examined by physics with pointing out possible material systems that fulfill the required qualities (comp. [5]). Moreover, the descriptions of the usage of such systems to create computing systems [15] have been proposed. Summing up, the next of the analysed theories, the one which is regarded to be valid nowadays, allows conducting an infinite number of operations in limited time a precisely chosen observer.

The above results do not entitle us to accept the thesis that hypercomputability is possible in our world. They show, however, that the possibility of crossing the Turing machine barriers is, in the light of some physical theories, real.

As we can observe, a new cognitive situation is introduced. The boundaries of computability become valid only for a stated physical theory. Moreover they receive provisional and temporary character. When a physical theory regarded as the proper description of the Universe changes, there may occur a change in computability boundaries. The theory of computability gained additionally a relative character, this time in relation towards the physical theory assumed as a starting point in a construction of computing systems.

Conclusions

We shall try to determine the conclusions arising from the discussed results. From a mathematical point of view the structure of certain computability model is based on the qualification of a means of computability procedure construction. Though a description built on a finite dictionary results in a limited number of such procedures, other restrictions are not connected with it. The examples of models given above and exceeding Turing's limits prove that a formal description of models, considerably different from classical models in their computing power, is possible. Of course, we can built many of such models with their cardinality not exceeding $\aleph_0$. That is why an explanation of the choice of one has to arrive from a domain beyond mathematics.

Because the aim of the computability theory is a description of qualities of mechanically computable procedures, it is natural to turn to physics. It is physics that allows us to determine which of the proposed models are physically realized (which devices can be really constructed). Thus the criterion of differentiating problems into computable and noncomputable is moved to the domain of empiricism. The problems that solving can be described in the reality of the material world, may be regarded as computable in a completely intuitive way. Such a classification could become an absolute one. It means that the structure of the Universe separates explicitly the range of possibly computable problems from noncomputable issues, with no relativity possible.

However, the cognitive situation is different. Namely, we have no direct knowledge of the quality of the Universe in order to carry out the above-mentioned computability classification in an unquestionable way. Our perception of the material world is limited by an intermediary factor – a physical theory. Different aspects of reality can be contained in different theories, sometimes even in the same segment of the world several various theories are permitted, provided they are empirically consistent. An additional factor of theoretical physics variability is time that brings new research paradigms.

In this contest the cognition of the degree of computability of a certain issue must be perceived through a commonly approved physical theory. This new situation changes the way of thinking within the theory of computability. The problems absolutely and simply computable are out of the question. Now the relationship between a problem and some physical theory should be pointed out. A problem P can be computable in relation towards quantum mechanics and noncomputable to Newton's mechanics.

Let us notice that such a view on the theory of computability refers not only to differentiation of problems into computable and noncomputable. The relativity towards a physical theory influences also the issues of complexity. It is worth pointing out that an adoption of a factor of time to be a continuous parameter allows in appropriate models to solve any number of operations (in suitably short time) in constant time. So the practical complexity (realization time, not a number of steps) of problems that are regarded to be intractable due to their classically understood complexity, may be reduced in this approach.

What is then the role of mathematics and logic in the theory of computability? Traditionally the sciences will act as the language that expresses research issues in this field. Moreover, protecting cohesion and completeness of introduced models, they will guarantee their correctness. But a choice of model and establishing limits of computing power will stay beyond the domain of mathematics, which is merely a research tool now.

The end of this paper will be presented in the form of concise thesis which results from all the above considerations.

The issue of computability is more a problem of ontology than epistemology. *It is a shape of physical reality, not any constructions a priori of human reason, that decides which problems are computable.*

Researches of the limits of computability are relative towards intermediary physical theories. Therefore, having no direct insight into the Universe ontology, *we have to base our researches of the computability theory on comprehensive physical theories.*

Considering computations beyond the limits of the Turing model is justified as a reasonable research program. The given examples of physical theories allowing hypercomputability (Newton's mechanics, the general relativity theory) show that exceeding the limits designed by a Turing machine model is not – at least potentially – unlikely.[2]

References

[1] H. P. Barendrengt. *The Lambda Calculus*, North-Holland, 1981.
[2] B. J. Copeland. The Church-Turing Thesis. In J. Perry and E. Zalta, (eds), *Stanford Encyclopedia of Philosophy*, 1996.

[2] The question of noncomputability in nature is a very controversional one. We have results of Pour-El and Richards [12] which suggest an existance of some physical phenomena beyond Turing computability. Contrary, others (e.g. [11]) reject a possibilty of noncomputable devices in nature pointing out an artificial and nonsmooth character of the mentioned examples.

[3] B. J. Copeland. Even Turing machines can compute uncomputable functions. In C. S. Calude, J. Casti, and M. J. Dinneen, (eds), *Unconventional Models of Computation*, Springer-Verlag, 1998.
[4] R. O. Gandy. Church's Thesis and principles of mechanisms. In J. Barwise, J. Keisler and K. Kunen, (eds), *The Kleene Symposium*, North-Holland, 1980.
[5] G. Etesi and I. Nemeti. Non-Turing computations via Malament-Hogarth space-times. *Int. J. Theor. Phys.* 41:341-370, 2002.
[6] A. A. Markow. Theory of algorithms (in Russian). *Trudy Matiematiczeskogo Instituta Stieklova*, 38: 176:189, 1951.
[7] C. Moore. Recursion theory on the reals and continuous-time computation. *Theoretical Computer Science*, 162:23-44, 1996.
[8] R. Murawski. *Funkcje rekurencyjne i elementy metamatematyki*, Wydawnictwo Naukowe UAM, 1991.
[9] J. Mycka, J. F. Costa. Real recursive functions and their hierarchy. Submitted to *Journal of Complexity*, 2003.
[10] P. Odifreddi. *Classical Recursion Theory*, North-Holland, 1989.
[11] R. Penrose. *The Emperor's New Mind*, Oxford University Press, 1989.
[12] M. B. Pour-El, J. I. Richards. Noncomputability in analysis and physics. *Adv. Math.* 48:44-74, 1983.
[13] L. A. Rubel. Some mathematical limitations of the general-purpose analog computer. *Advances in Applied Mathematics*, 9:22-34, 1988.
[14] L. A. Rubel. The extended analog computer. *Advances in Applied Mathematics*, 14:39-50, 1993.
[15] O. Shagrir, I. Pitowsky. Physical Hypercomputation and the Church-Turing Thesis. *Minds and Machines*, 13: 87-101, 2003.
[16] C. Shannon. Mathematical theory of the differential analyzer. *J. Math. Phys. MIT*, 20:337-354, 1941.
[17] R. Soare. The history and concept of computability. In E. Griffor, (ed), *Handbook of Computability Theory*, Elsevier, 1999.
[18] A. Turing. On computable numbers, with application to the Entscheidungsproblem. *Proc. London Math. Soc.*, Ser. 2, 42: 230-265, 1936.
[19] Z. Xia. The existence of noncollision singularities in Newtonian systems. *The Annals of Mathematics*, 135(3):411-468, 1992.

Jerzy Mycka
Institute of Mathematics
University of Maria Curie-Sklodowska
Lublin, Poland